

# **REPORT**

## **ZDEMO\_INTERCTIVE\_RPRTS\_BUTTONS**

### **NO STANDARD PAGE HEADING LINE-SIZE**

### **150 .**

types : begin of ty\_lfa1 , " from vender master  
table(general section)

lifnr type lifnr , " Account Number of Vendor or  
Creditor

name1 type name1\_gp, " name

ort01 type ort01\_gp, " city

land1 type land1\_gp, " country key

end of ty\_lfa1.

types : begin of ty\_ekko , " Purchasing Document  
Header

ebeln type ebeln , "Purchasing Document Number

aedat type erdat , "Date on Which Record Was  
Created

LIFNR TYPE LIFNR,

end of ty\_ekko.

data : it\_lfa1 type table of ty\_lfa1,

it\_ekko type table of ty\_ekko,

wa\_lfa1 type ty\_lfa1,

wa\_ekko type ty\_ekko,

v\_lifnr type lifnr,

V\_NAME1 TYPE NAME1\_GP,

IT\_SELECTED\_LFA1 TYPE TABLE OF TY\_LFA1,  
v\_box(1) type c.

select-options s\_lifnr for v\_lifnr default 1 to 100.

select lifnr name1 ort01 land1 from lfa1 into table  
it\_lfa1 where lifnr in s\_lifnr.

SET PF-STATUS 'ZSTATUS'.  
perform display\_data.

sy-lsind = 0.

AT USER-COMMAND.  
CASE SY-UCOMM.

WHEN 'SEAL'.  
V\_BOX = 'X' . " CHECKED.  
PERFORM DISPLAY\_DATA.  
WHEN 'DSEAL'.  
V\_BOX = ' '. "DESELECT  
PERFORM DISPLAY\_DATA.  
WHEN 'DOWN'.  
REFRESH IT\_SELECTED\_LFA1.  
DATA V\_LINES TYPE i.

DESCRIBE LIST NUMBER OF LINES V\_LINES.

```

DO V_LINES TIMES.
CLEAR WA_LFA1.
READ LINE SY-INDEX FIELD VALUE V_BOX WA_LFA1-
LIFNR.

IF V_BOX = 'X'.
CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
EXPORTING
  INPUT      = WA_LFA1-LIFNR
IMPORTING
  OUTPUT     = WA_LFA1-LIFNR
.
READ TABLE IT_LFA1 INTO WA_LFA1 WITH KEY lifnr =
WA_LFA1-LIFNR.

IF SY-SUBRC = 0.
APPEND WA_LFA1 TO IT_SELECTED_LFA1.
ENDIF.
CLEAR : WA_LFA1, V_BOX.
ENDIF.
ENDDO.

IF IT_SELECTED_LFA1 IS NOT INITIAL.
CALL METHOD
CL_GUI_FRONTEND_SERVICES=>GUI_DOWNLOAD
EXPORTING
  FILENAME = 'D:/BUTTON/BUTTON.TXT'
  write_field_separator = ' '
CHANGING
  DATA_TAB = IT_SELECTED_LFA1.

```

```
IF SY-SUBRC = 0.  
  MESSAGE 'SUCCESSFUL DOWNLOADED' TYPE 'S'. "  
STATUS MESSAGE  
ENDIF.
```

```
ELSE.  
  MESSAGE 'NO DATA TO DOWNLOAD ' TYPE 'S'. "  
STATUS MESSAGE  
ENDIF.
```

```
.
```

```
WHEN 'GETPOS'.  
  PERFORM COLLECT_SELECTED_LINES.
```

```
IF IT_SELECTED_LFA1 IS NOT INITIAL .
```

```
  SELECT EBELN AEDAT LIFNR INTO TABLE IT_EKKO  
  FROM EKKO FOR ALL ENTRIES IN IT_SELECTED_LFA1  
  WHERE LIFNR = IT_SELECTED_LFA1-LIFNR.
```

```
  IF IT_EKKO IS NOT INITIAL.  
    SORT IT_EKKO BY LIFNR EBELN.  
    LOOP AT IT_EKKO INTO WA_EKKO.  
      WRITE :/ SY-TABIX, WA_EKKO-EBELN, WA_EKKO-LIFNR,  
WA_EKKO-AEDAT.  
    ENDLOOP.  
  ELSE.  
    WRITE :/ 'NO PURCHASE FOUND FROM VENDER'.
```

ENDIF.

ELSE.

MESSAGE 'NO VENDER SELECTED FROM THE LIST '  
TYPE 'S'.  
ENDIF.

ENDCASE.

FORM DISPLAY\_DATA.

LOOP AT it\_lfa1 into wa\_lfa1.

write :/ V\_BOX AS CHECKBOX , wa\_lfa1-lifnr color 1 ,  
wa\_lfa1-name1 color 3 , wa\_lfa1-ort01 color 5, wa\_lfa1-  
land1 color 7.

ENDLOOP.

ENDFORM.

FORM COLLECT\_SELECTED\_LINES .

REFRESH IT\_SELECTED\_LFA1.

DATA V\_LINES TYPE I.

DESCRIBE LIST NUMBER OF LINES V\_LINES.

DO V\_LINES TIMES.

CLEAR WA\_LFA1.

READ LINE SY-INDEX FIELD VALUE V\_BOX WA\_LFA1-  
LIFNR.

```
IF V_BOX = 'X'.  
  CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'  
  EXPORTING  
    INPUT      = WA_LFA1-LIFNR  
  IMPORTING  
    OUTPUT     = WA_LFA1-LIFNR  
  .  
  READ TABLE IT_LFA1 INTO WA_LFA1 WITH KEY LIFNR =  
  WA_LFA1-LIFNR.  
  
  IF SY-SUBRC = 0 .  
    APPEND WA_LFA1 TO IT_SELECTED_LFA1.  
  
  ENDIF.  
  CLEAR : WA_LFA1, V_BOX.  
  ENDIF.  
  ENDDO.  
ENDFORM.
```